Module 8

Key Distribution

CRYPTOGRAPHY

Ana I. González-Tablas Ferreres

José María de Fuentes García-Romero de Tejada

Lorena González Manzano

Pablo Martín González

Sergio Pastrana Portillo

uc3m | Universidad Carlos III de Madrid

COSEC

OUTLINE

- 8. Key distribution and asymmetric encryption
  - Key distribution
    - Symmetric key distribution using symmetric cryptography
    - Symmetric key distribution using asymmetric cryptography --- Hybrid cipher (KEM/DEM)
    - Distribution of public keys
    - Key exchange protocols: Diffie-Hellman

OUTLINE

- 8. Key distribution and asymmetric encryption
  - Key distribution
    - **Symmetric key distribution using symmetric cryptography**
    - Symmetric key distribution using asymmetric cryptography --- Hybrid cipher (KEM/DEM)
    - Distribution of public keys
    - Key exchange protocols: Diffie-Hellman

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Symmetric key distribution using symmetric cryptography

- Secret key cryptosystems require that sender and receiver share a priori a secret key

- Problem: How to share/distribute secret keys in a secure way?

- Let's see the possibilities of how we may solve it by only using symmetric cryptography

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Symmetric key distribution using symmetric cryptography

- Possibilities:
  1. A generates a key and hands in it to B (in person)
  2. A third party chooses the key and hands in it to both A and B (in person)
  3. If A and B already share a key, they can use it to encrypt a new key and share it with the other party
  4. If A and B have a secure channel with a third party C, C can choose the key and (securely) share it with both A and B

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

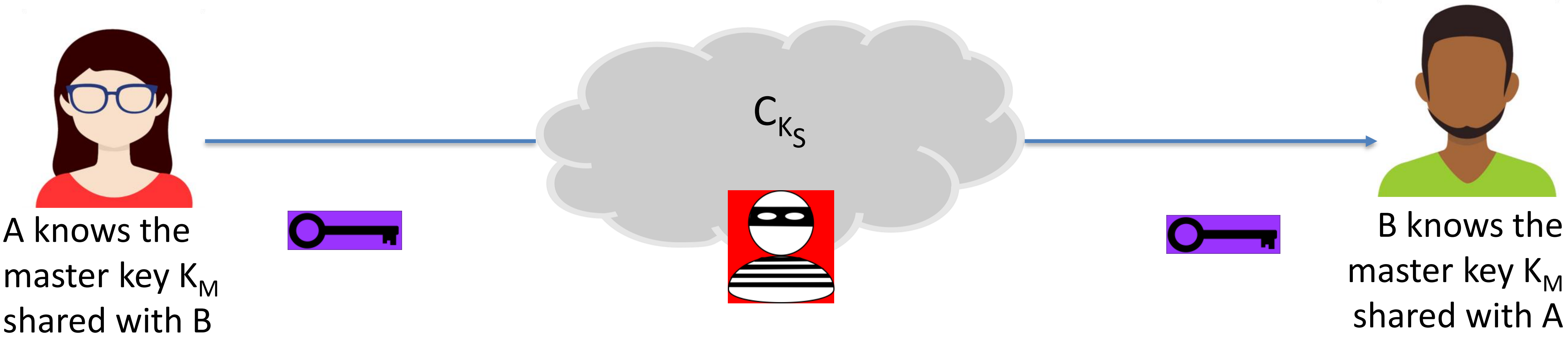
Symmetric key distribution using symmetric cryptography

- Possibilities:
  1. A generates a key and hands in it to B (in person)
  2. A third party chooses the key and hands in it to both A and B (in person)
  3. If A and B already share a key, they can use it to encrypt a new key and share it with the other party
  4. If A and B have a secure channel with a third party C, C can choose the key and (securely) share it with both A and B

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Symmetric key distribution using symmetric cryptography

- Key wrapping = Encrypting a symmetric key with another symmetric key



A chooses K_S, a symmetric session key

A encrypts K_S using K_M, the master key shared with B, and sends it to B

$$C_{K_S} = E_{SIM}(K_M, K_S)$$

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Symmetric key distribution using symmetric cryptography

- Possibilities:
  1. A generates a key and hands in it to B (in person)
  2. A third party chooses the key and hands in it to both A and B (in person)
  3. If A and B already share a key, they can use it to encrypt a new key and share it with the other party
  4. If A and B have a secure channel with a third party C, C can choose the key and (securely) share it with both A and B

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Symmetric key distribution using symmetric cryptography

- Key hierarchy with a KDC (Key Distribution Center)
  - Each user shares a master key K_X with the KDC
    - Keys K_X are used to encrypt one-time session keys K_S , being then delivered to the users by the KDC
    - A and B want to securely communicate between them
    - The KDC creates a one-time session key K_S and delivers it encrypted for A and B using the master keys the KDC shares with those users (K_A and K_B)
    - A and B use K_S to encrypt the data exchanged between them
  - Number of keys needed for n users:
    - n · (n - 1)/2 session keys (simultaneously)
    - n master keys
  - It is necessary that users previously share the master key with the KDC

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

OUTLINE

- 8. Key distribution and asymmetric encryption
  - Key distribution
    - Symmetric key distribution using symmetric cryptography
    - **Symmetric key distribution using asymmetric cryptography --- Hybrid cipher (KEM/DEM)**
    - Distribution of public keys
    - Key exchange protocols: Diffie-Hellman

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Symmetric key distribution using asymmetric cryptography --- Hybrid cipher (KEM/DEM)

Symmetric (secret key)

- Pros
  - Symmetry
  - Fast
- Cons
  - Need a secure channel (to exchange the key)
  - Difficult management of a high number of keys

Asymmetric (public key)

- Cons
  - Asymmetry
  - Slow
- Pros
  - They do not need a secure channel to exchange the public key
  - ***"Easy"*** management of a high number of keys

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Symmetric key distribution using asymmetric cryptography --- Hybrid cipher (KEM/DEM)

- Both types have some important problem
  - Asymmetric cryptosystems are really slow (compared to symmetric ones)
  - Symmetric cryptosystems need a secure channel to distribute the keys, and key management is challenging
- Solution:
  - Use asymmetric cryptography to distribute symmetric keys
  - Use symmetric cryptography to encrypt data
  - This combinations is known as **hybrid encryption** or **KEM/DEM**
    - **Key Encapsulation Mechanism (KEM)** – "asymmetric part"
    - **Data Encapsulation Mechanism (DEM)** – "symmetric part"

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Symmetric key distribution using asymmetric cryptography --- Hybrid cipher (KEM/DEM)

- Cleartext M is encrypted with a symmetric cipher (eg., AES) using session key K_S , that is randomly generated at that moment
- Session key K_S is asymmetrically encrypted (eg., RSA) using the public key of the receiver $K_{U,B}$

- Receiver decrypts first the session key K_S, using his/her private key
- Then using the session key K_S, decrypts the encrypted message

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Symmetric key distribution using asymmetric cryptography --- Hybrid cipher (KEM/DEM)

A chooses K_S, symmetric session key

A encrypts M using K_S and sends the ciphertext to B

$$C_M = E_{SIM}\ (K_S,\ M)$$

A encrypts K_S using B's public key $K_{U,B}$, and sends the ciphertext to B

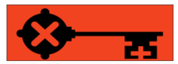
$$C_{K_S} = E_{ASIM}(K_{U,B},\ K_S)$$

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Symmetric key distribution using asymmetric cryptography --- Hybrid cipher (KEM/DEM)

B first decrypts C_{K_S} using his private key $K_{V,B}$ to obtain K_S

$K_S = D_{ASIM}(K_{V,B}, C_{K_S})$ → $K_{U,B}, K_{V,B}, K_S$



Then B decrypts decrypts C_M using Ks to obtain M

$M = D_{SIM}(K_S, C_M)$ → $K_{U,B}, K_{V,B}, K_S, M$

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Symmetric key distribution using asymmetric cryptography --- Hybrid cipher (KEM/DEM)

- RSA-KEM
  - Let’s define a KEM using RSA encryption/decryption function and a hash function H to enhance the overall security of the scheme
  - A chooses $x \in \{1,\ldots,n_B-1)$, and encrypts this number for B using his public key: $c = x^{e_B}$ mod. n_B. A also computes the symmetric key that we’ll be used for securely communicating with B: $k = H(x)$.
  - A sends c to B
  - B decrypts c using his private key: $x = x^{d_B}$ mod. n_B. Then, computes the shared key the same way as A did: $k = H(x)$

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

OUTLINE

- 8. Key distribution and asymmetric encryption
  - Key distribution
    - Symmetric key distribution using symmetric cryptography
    - Symmetric key distribution using asymmetric cryptography --- Hybrid cipher (KEM/DEM)
    - **Distribution of public keys**
    - Key exchange protocols: Diffie-Hellman

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Distribution of public key

- Possibilities:
  1. Public announcement
  2. Publicly available directory
  3. Public-key authority
  4. Public-key certificates (public-key certification authorities)

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Distribution of public key. Publicly available directory

- Public key directory

- Properties:
  - Entries of the type: {name, public key}
  - Secure registration (in person or using a secure authenticated channel)
  - Users may replace a key at any time
  - Access to the directory entries can be in person or using some communication network (through an authenticated communication channel from the directory to the users)

- Directory needs to be trusted

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

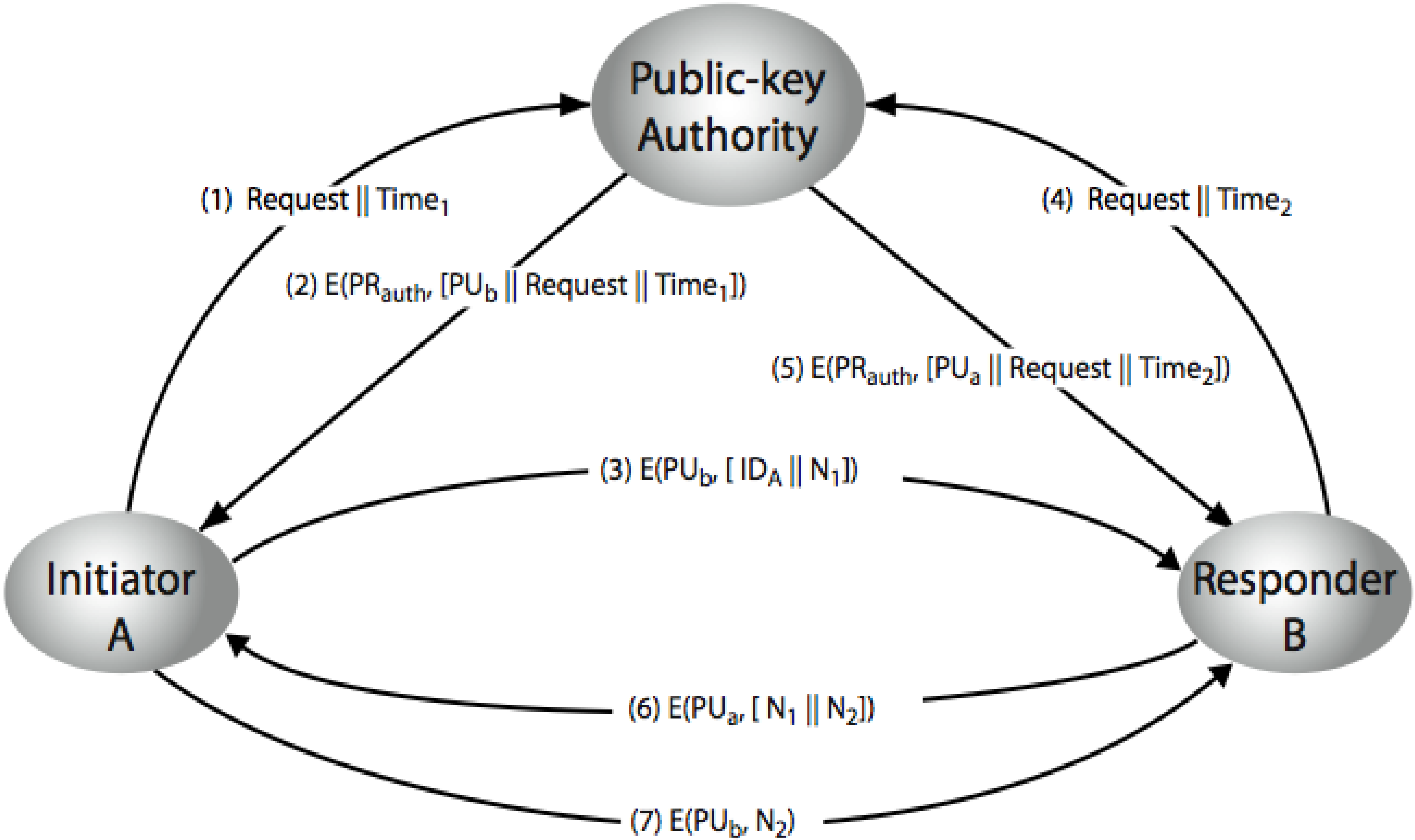
Distribution of public key. Public-key authority

- Properties similar to a public key directory but with more control mechanisms over the public keys
- Requires that participants know the public key of the public-key authority
- Requires online access (in real time) to the public-key authority (probable bottle-neck problem)

- Protocols similar to the one depicted in next slide are used

COSEC uc3m
Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Distribution of public key. Public-key authority



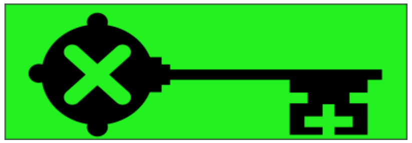
Source of the Figure: Cryptography and Network Security. Principles and Practices, 5th ed., 2011. Pearson Education.

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Distribution of public key.
Public-key certification authorities

- Public-key Certification Authority (CA) issues public-key certificates
- Public-key certificates allow public key distribution with an offline authority (avoiding bottle-neck problem)
- A public key certificate binds in a secure way (authenticity, integrity) a public key and an identity

A **public key**



An **identity**

Validity period

Usage rights…

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Distribution of public key. Public-key certification authorities

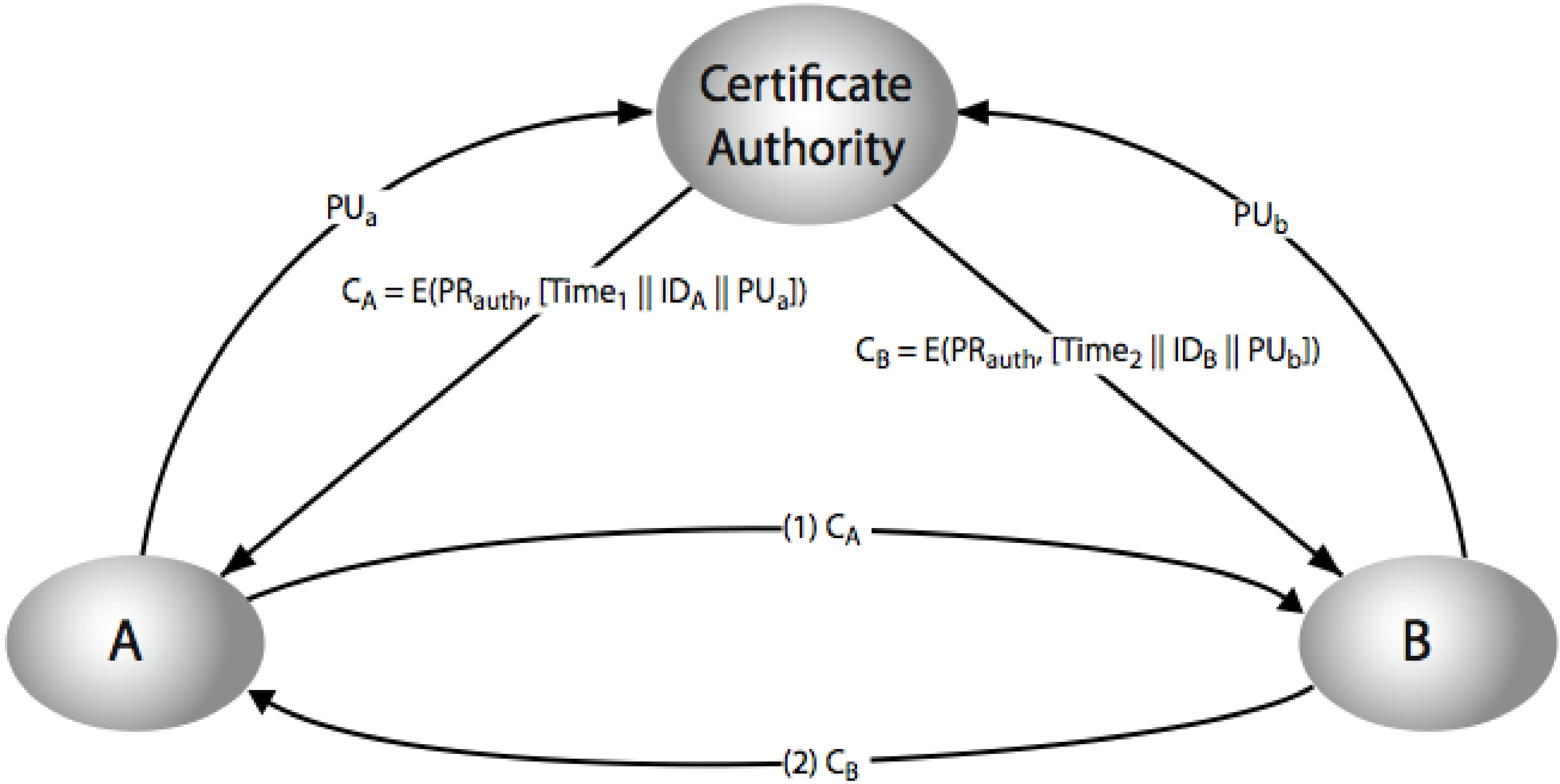
- Pubic key certificates are signed by a Certification Authority (CA) [*we'll study digital signatures later*]

- Validity of public key certificates can be verified by anyone knowing the CA's public key

- Main idea: if we trust the CA to correctly bind an identity to a public key, we'll trust the certificates she has issued
  - Conditioned to the correct verification of the certificate

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Distribution of public key.
Public-key certification authorities



Source of the Figure: Cryptography and Network Security. Principles and Practices, 5th ed., 2011. Pearson Education.

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Distribution of public key. Public-key certification authorities

How do we sign a message?
How do we verify the signature on a message?

Is the CA's public key certified? Who certifies it?

We'll study digital signatures, public key certificates and Public Key Infrastructures soon

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

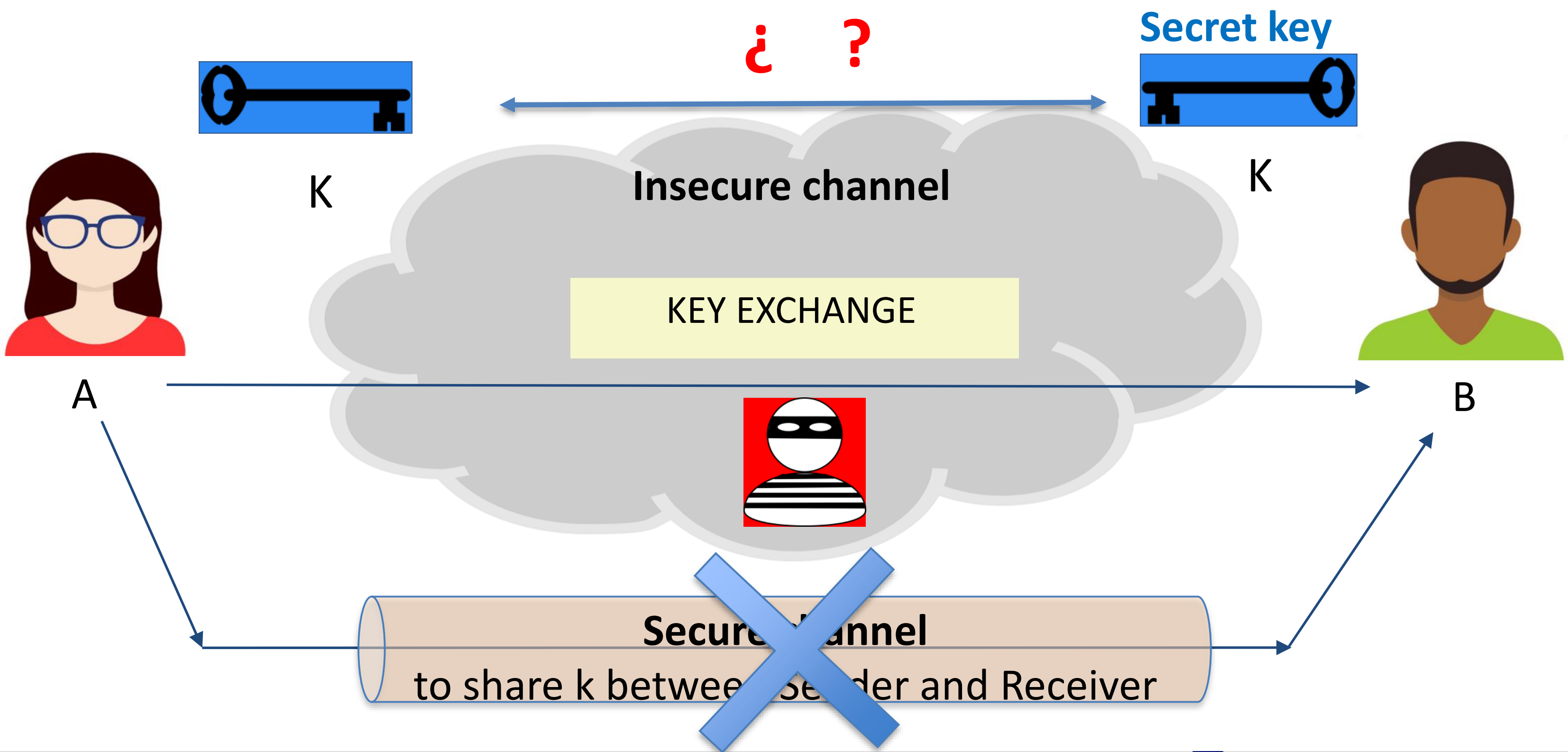
OUTLINE

- 8. Key distribution and asymmetric encryption
  - Key distribution
    - Symmetric key distribution using symmetric cryptography
    - Symmetric key distribution using asymmetric cryptography --- Hybrid cipher (KEM/DEM)
    - Distribution of public keys
    - **Key exchange protocols: Diffie-Hellman**

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Key exchange protocols: Diffie-Hellman

- Problem:
  – Two parties, who have not shared a priori a secret, must exchange a secret over an insecure channel



Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Key exchange protocols: Diffie-Hellman

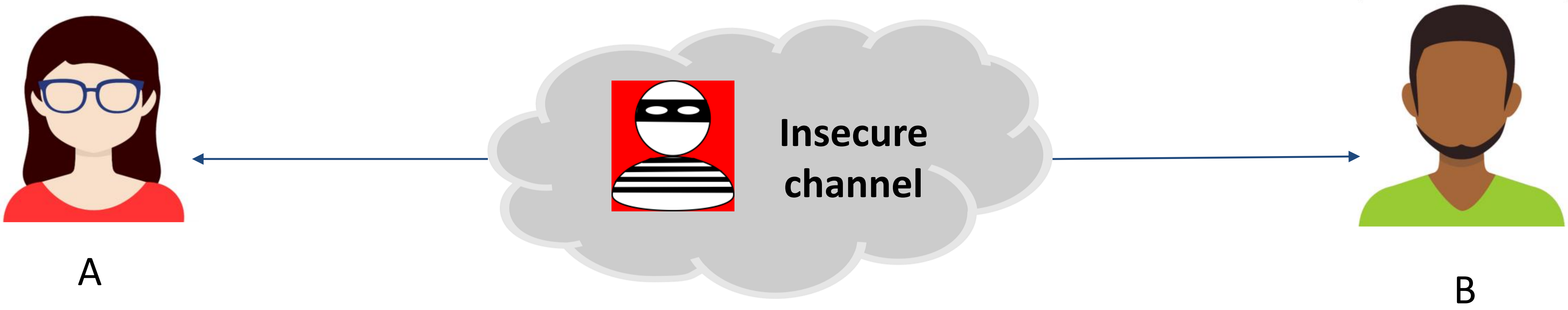
- Symmetric cryptography does not solve this problem
- Public key cryptography does:
  - Public key cryptography uses trapdoor one-way functions, easy to compute in one direction but *very hard* to compute for anyone that does not know the "trapdoor"
  - Public key cryptography allows to make public one parameter (the public key/part) while making *very hard* to infer a second parameter that is kept private (the private key/part or trapdoor)
- The Diffie-Hellman protocol allows two entities to exchange a symmetric key through a public channel using public key cryptography

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Key exchange protocols: Diffie-Hellman

- Whitfield Diffie, Martin E. Hellman. ***New Directions in Cryptography***. IEEE Transactions in Information Theory, v. IT-22, pp 664-654. November 1976.
  - Seminal article that proposed public key cryptography
  - Probably the biggest cryptographic milestone in 3,000 years
  - It was previously discovered by British Intelligence Services
  - It proposes asymmetric cryptosystems --- in a theoretical way --- and the **Diffie-Hellman key exchange algorithm**, based also in asymmetric cryptography

Key exchange protocols: Diffie-Hellman



A chooses p, very large prime number, and g, generator of GF(p)

A chooses $x_A \in G(p)$, private parameter of A or ephemeral secret, random | $1 < x_A < p - 1$

A computes y_A , ephemeral public key of A ($y_A = g^{x_A}$ mod. p), and sends it to B along g and p

p, g, y_A →

B chooses $x_B \in GF(p)$, private parameter of B or ephemeral secret, random | $1 < x_B < p - 1$

B computes y_B , ephemeral public key of B ($y_B = g^{x_B}$ mod. p), and sends it to A

← y_B

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Key exchange protocols: Diffie-Hellman

A computes $K = y_B^{\ x_A}$ mod. p

B computes $K = y_A^{\ x_B}$ mod. p

Both have computed the same symmetric key:

$K = y_B^{\ x_A}$ mod. p = $(g^{x_B})^{x_A}$ mod. p = $g^{x_B \cdot x_A}$ mod. p

$K = y_A^{\ x_B}$ mod. p = $(g^{x_A})^{x_B}$ mod. p = $g^{x_A \cdot x_B}$ mod. P

Once they have agreed on a symmetric key K, they can use it to secure their communications

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Key exchange protocols: Diffie-Hellman

- Security of the Diffie-Hellman key exchange protocol is based on:
  - Computing x (the private part) knowing only y (the public part) is *very hard* (computationally). It is known as the discrete logarithm problem
  - Computing K knowing only y_A and y_B is also *very hard* computationally. It is known as the Diffie-Hellman problem

- In practice:
  - Parameters p and g are standardized and are known by everybody
  - K is not used as symmetric key directly, it is necessary to derive another symmetric key K' or set of keys that have more entropy and satisfy other security requirements
    - Eg., a naïve way to derive K' is using a hash function: K' = H (K)

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Key exchange protocols: Diffie-Hellman

- Vulnerabilities
  - (Anonymous) Diffie-Hellman protocol is **not secure** against active adversaries, as there is no authentication in the exchanged messages
  - It is vulnerable to *Person In The Middle** attacks
    - The adversary Mallory controls the channel
    - Mallory impersonates A when communicating with B, and also impersonates B when communicating with A
    - Mallory performs a Diffie-Hellman key exchange with both A and B
    - Neither A nor B notice they are communicating with Mallory instead of with B or A
  - SOLUTION: Authenticate the exchanged parameters (ephemeral public key) binding them to an identity by signing them

* Known till now as *Man in the Middle*

Ana I. González-Tablas Ferreres, José M. de Fuentes García-Romero de Tejada, Lorena González Manzano, Pablo Martín González, Sergio Pastrana Portillo

Key exchange protocols: Diffie-Hellman

- There is an elliptic curve version of the Diffie-Hellman key exchange protocol
- It works on a cyclic group defined on the elliptic curve
  - Similar to the cyclic group obtained when computing the powers of an integer modulo n when the integer is a generator of that group
  - It is necessary to select a prime p, an elliptic curve and a primitive point P in the curve that works as "generator"
  - The "generation" operation is the multiplication by an integer (repeated addition of the primitive point)
    - P, 2P, 3P, 4P, 5P…
  - A selects secret key x_A and computes public key as $Y_A = x_A \cdot P$
  - B selects secret key x_B and computes public key as $Y_B = x_B \cdot P$
  - A sends Y_A to B and B sends Y_B to A
  - Both compute the shared key as $K = x_A \cdot x_B \cdot P$

CRYPTOGRAPHY

COSEC

uc3m | Universidad Carlos III de Madrid